METHODS AND TOOLS FOR ANALYSIS OF SYMMETRIC CRYPTOGRAPHIC PRIMITIVES

Oleksandr Kazymyrov

DISSERTATION FOR
THE DEGREE OF PHILOSOPHIAE DOCTOR

THE SELMER CENTER
DEPARTMENT OF INFORMATICS
UNIVERSITY OF BERGEN
NORWAY

DECEMBER 1, 2014

ACKNOWLEDGMENTS

It is impossible to thank all those who have, directly or indirectly, helped me with this thesis, giving of their time and experience. I wish to use this opportunity to thank some of them.

Foremost, I would like to express my very great appreciation to my main supervisor Tor Helleseth, who has shared his extensive knowledge and experience, and made warm conditions for the comfortable research in one of the rainiest cities in the world. I owe a great deal to Lilya Budaghyan, who was always ready to offer assistance and suggestions during my research. Advice given by Alexander Kholosha has been a great help in the early stages of my work on the thesis.

My grateful thanks are also extended to all my friends and colleagues at the Selmer Center for creating such a pleasant environment to work in. I am particularly grateful to Kjell Jørgen Hole, Matthew G. Parker and Håvard Raddum for the shared teaching experience they provided. Moreover, I am very grateful for the comments and propositions given by everyone who proofread my thesis. In addition, I would like to thank the administrative staff at the Department of Informatics for their immediate and exhaustive solutions of practical issues.

I wish to acknowledge the staff at the Department of Information Technologies Security, Kharkiv National University of Radioelectronics, Ukraine, especially Roman Oliynykov, Ivan Gorbenko, Viktor Dolgov and Oleksandr Kuznetsov for their patient guidance, enthusiastic encouragement and useful critiques.

I would like to offer my special thanks to my wife, who made invaluable contributions, including reading of the early versions of the thesis and making extremely beneficial and penetrating observations on the research results.

Abstract

The development of modern cryptography is associated with the emergence of computing machines. Since specialized equipment for protection of sensitive information was initially implemented only in hardware, stream ciphers were widespread. Later, other areas of symmetric and asymmetric cryptography were established with the invention of general-purpose processors. In particular, such symmetric cryptographic primitives as block ciphers, message authentication codes (MACs), authenticated ciphers and others began to develop rapidly. Today various cryptographic algorithms are commonly used in everyday life to protect private data.

Design and analysis of advanced symmetric cryptographic primitives require a lot of time and resources. This is related to many factors, mainly to the cryptanalysis of prospective encryption algorithms under development. Every year new and modified attacks are published, leading to a rapid increase in the quantity of requirements and criteria imposed on cryptoprimitives.

Most of this thesis is devoted to analysis and improvement of cryptographic attacks and corresponding criteria for basic components. Almost all modern cryptoprimitives use nonlinear mappings for protection against advanced attacks. In connection with that a new method was proposed for the generation of random substitutions (S-boxes) with extreme cryptographic indicators that can be used in the next-generation ciphers to provide high and ultra-high security levels. In addition, several criteria imposed on S-boxes used in block ciphers were analyzed and their significance for block ciphers was proven. It is worth mentioning a practical method of testing two vectorial Boolean functions and a universal tool for checking properties of arbitrary binary nonlinear components presented in papers gathered in this thesis.

Another part of the thesis is dedicated to the cryptanalysis of hash functions as well as block and stream ciphers. To be more precise, an algebraic attack based on a binary decision diagram (BDD) was performed on the reduced Data Encryption Standard (DES), a scaled-down version of Advanced Encryption Standard (AES) and extended affine (EA) equivalence problem. Moreover, an algebraic approach was used to reconstruct an initial representation of the current Russian hash standard GOST 34.11-2012. Finally, a backward states tree method has been used to analyze stream ciphers based on the combination principle of linear and nonlinear feedback registers.

List of Papers

[I] Kazymyrov, O., Raddum, H.: Algebraic attacks using binary decision diagrams. *In Pre-proceedings of BalkanCryptSec 2014*, pp. 31–44, 2014.

[II] Eilertsen, A. M., Kazymyrov, O., Kazymyrova, V., Storetvedt, M.: A Sage library for analysis of nonlinear binary mappings. In *Pre-proceedings of Central European Conference on Cryptology (CECC14)*, pp. 69–78, 2014.

[III] Kazymyrov, O., Kazymyrova, V., Oliynykov, R.: A method for generation of high-nonlinear S-boxes based on gradient descent. In *Mathematical Aspects of Cryptography*, vol. 5, pp. 71–78. Steklov Mathematical Institute, 2014.

[IV] Kazymyrov, O., Kazymyrova, V.: Algebraic aspects of the Russian hash standard GOST R 34.11-2012. In *Pre-proceedings of 2nd Workshop on Current Trends in Cryptology (CTCrypt 2013)*, pp. 160–176, 2013.

[V] Kazymyrov, O., Kazymyrova, V.: Extended criterion for absence of fixed points. In *Pre-proceedings of 2nd Workshop on Current Trends in Cryptology (CTCrypt 2013)*, pp. 177–191, 2013.

[VI] Helleseth, T., Jansen, C.J.A., Kazymyrov, O., Kholosha, A.: State space cryptanalysis of the MICKEY cipher. In *Information Theory and Applications Workshop (ITA)*, pp. 1–10. Institute of Electrical and Electronics Engineers (IEEE), 2013.

[VII] Budaghyan, L., Kazymyrov, O.: Verification of restricted EA-equivalence for vectorial Boolean functions. In Özbudak, F., Rodríguez-Henríquez, F. (eds.), *Arithmetic of Finite Fields*, vol. 7369 of *Lecture Notes in Computer Science*, pp. 108–118. Springer Berlin Heidelberg, 2012.

Contents

Scientific Results

Introduction

1. THE STATE-OF-THE-ART OF SYMMETRIC CRYPTOLOGY

One of the strategic priorities of any country is to adopt comprehensive measures to protect the national information space [1]. The main feature of this trend is to increase performance and to improve security in telecommunication systems. Fast and secure access to information and computing resources, most of which are a part of the Internet, may be regarded as one of the requirements of a developed country.

Information technologies (IT) are an essential part of our daily lives. Efficiency of application and operation of information systems depend on their security and reliability. There are many fields where unpredictable or abnormal operation of telecommunication systems may result in serious consequences. These include management and control systems of water, gas and energy supply; petroleum and nuclear industries; transport systems, etc. Over the past few decades the number of publications and projects related to different aspects of information security has considerably increased.

The emergence of new problems requires improved methods to solve them [2, 3]. Until recently, cryptographic tools were available only to special state authorities. Today they are used in everyday life in the process of creating, sending, receiving, processing, storing and destroying data [3, 4].

Block ciphers play an important role in complex information systems [3, 5]. They are widely used due to their high efficiency and low implementation complexity. In addition to providing confidentiality, block ciphers realize message authentication codes (MACs), hash functions, pseudorandom number generators (PRNG) and authentication protocols [3, 6]. Thus, block ciphers are used in most modern symmetric cryptographic primitives. Nonetheless, special algorithms have many advantages. For example, to provide secure high-speed transmission of information, especially when the data processing is in hardware, stream ciphers are used. Due to their structure they are optimized for hardware platforms by default. At the same time their performance can be ten times better than of block ciphers.

Many international competitions for choosing hash functions, block and stream ciphers have shown that the task of creating a secure cryptographic algorithm is rather complicated. For example, all stream ciphers submitted to the New European Schemes for Signatures, Integrity and Encryption (NESSIE) were theoretically broken [7]. At the same time, the role of the cryptographic community should not be underestimated. Every year more and more people invent new and advanced approaches to solve cryptographic problems.

In view of the above, the goal of this thesis is to improve the resistance of modern iterative cryptographic primitives to advanced attacks through the development of methods and tools of cryptanalysis.

1.1. Global development of symmetric cryptography

At the end of the 20th century a number of successful theoretical attacks allowed the block cipher DES to be broken [8]. A bit later practical implementations emerged to find the encryption key in a reasonable time [9]. As a consequence in the USA in 1997, the Advanced Encryption Standard (AES) competition was launched [10]. The main objective of the competition was the selection of a new generation block cipher as the standard. After several years of research the algorithm Rijndael was selected as a winner. This cipher was became the encryption standard FIPS-197, also known as AES [11]. Rijndael ranked first due to its high-level resistance against known attacks, simple description, and high performance on most platforms of that time.

A similar European open competition NESSIE was launched in February 2000 [5, 7]. The main task of the project was the selection of the best cryptographic primitives among submitted candidates from around the world. Security, performance, and flexibility were offered as the main criteria. After the competition a recommended list containing block ciphers, hash functions, MACs and digital signature algorithms for industrial usage was created [7].

Along with other cryptographic algorithms six stream ciphers were submitted to NESSIE [7]. All of them as mentioned above were theoretically broken. This led, in November 2004, to a separate project called eSTREAM, whose main task was to choose one or more stream ciphers for use in the business sector [12]. It should be mentioned that the stream ciphers were divided into two separate categories. While the first one consisted of software oriented primitives, another contained algorithms adapted for hardware applications. After four years of research, 4 ciphers were selected for each category. However, in 2008 the stream cipher F-FCSR-H v2 was excluded from the list because of vulnerabilities [13].

In parallel with NESSIE a similar research was carried out by the Japanese government under CRYPTREC [14]. As a result of this analysis the best algorithms were selected for data protection. As of today many cryptoprimitives have been recommended for use in both government (e.g., AES, Camellia, KCipher-2, ECDH, SHA-512, HMAC, etc.) and business (e.g., MISTY1, MUGI, SC2000, PC-MAC-AES, PSEC-KEM, etc.) sectors [15].

In post-Soviet states the block cipher GOST 28147 is used [16]. It was adopted in 1989 and has been outdone in performance, usability and other characteristics by modern ciphers, including AES. In the past few years theoretical attacks on this encryption algorithm have been successfully carried out. The complexity of finding the key was reduced from 2^{256} to 2^{225} [17, 18]. However, the complexity of 2^{225} is unachievable for modern computers so GOST 28147 remains practically secure[19].

However, long before the proposed attacks the cryptographic community and government agencies of these countries began to think about changing the encryption algorithm. Since 2003 Belarus has used a new standard for confidentiality and integrity [20]. It includes a block cipher and its modes of operation, a message authentication code and a hash function.

In order to find an alternative to GOST 28147 the State Service of Special Communication and Information Protection of Ukraine announced in 2006 an open competition to design a prototype of a block cipher for the new standard [21]. One of the main requirements for the prospective cipher was a high-level of resistance against known and promising types of cryptanalytic attacks. At the same time, it was necessary to achieve a performance not less than the previous standard. In practice the designers tried to beat AES. According to the results of the competition in 2009 the cipher Kalyna was allowed to be used for protection of nongovernmental information [3, 21]. This cipher with improvements is now undergoing a formal assessment, and is at the stage of adoption as the standard [22].

In November 2007 the National Institute of Standards and Technology (NIST) opened a competition to develop a hash function SHA-3, which would complement the existing two versions [23]. In analogy with AES, NIST teamed cryptanalysts and developers from around the world in order to select one or more additional hash algorithms. In October 2012 it was announced that SHA-3 will be based on the algorithm Keccak [24]. Two years later a draft version of a new standard was published [25].

Unlike the USA, Russia did not announce an open competition, and used the hash function Stribog (Streebog) as a prototype [26–28]. This algorithm is the only known version of the draft state standard. Since January 1st, 2013 GOST R 34.11-2012 came into effect, replacing the earlier version [29]. Further development of block ciphers in Russia was presented at CTCrypt'14 [30]. According to the article the current standard GOST 28147 will be used in hardware, and the new block cipher Kuznechik (Grasshopper) will target software.

A similar path has been chosen by Ukraine in the development of the hashing algorithm. Drawing on the experience gained from cryptanalysis of block ciphers and considering finished competitions, Grøstl was taken as a basis for the new hash function. Together with Keccak, Grøstl is one of five finalists of SHA-3 [24, 31]. The main difference of the Ukrainian hash function is the usage of Kalyna with 512-bit block and key length instead of AES in the functions P and Q [31]. As in the case of the block cipher, the hash function is at the final stage of the standardization procedure.

In recent years the question about improvement of methods providing security and integrity of transmitted data simultaneously has been increasingly raised. In connection with this, the Competition for Authenticated Encryption: Security, Applicability, and Robustness (CAESAR) was organized [32]. Over the next few years, cryptologists, and software and hardware specialists from all over the world will select a modern authenticated cipher.

1.2. General design ideas of cryptographic primitives

1.2.1. Block ciphers

Let $E : \{0,1\}^l \times \{0,1\}^k \mapsto \{0,1\}^l$ be a function which takes a key K of length k bits, an input message (plaintext) M of length l bits and returns an output message (ciphertext) $E(M,K)$. For each K let $E_K : \{0,1\}^l \mapsto \{0,1\}^l$ be a function defined by $E_K(M) = E(M,K)$. Then E is a block cipher if E_K and E_K^{-1} are efficiently computable and E_K is a permutation for every K.

Most modern block ciphers are iterative (Fig. 1). A round function is usually used multiple times with different parameters (round keys). An arbitrary iterative block cipher can be mathematically described as follows

$$E_K(M) = PW_{k_{r+1}} \circ \prod_{i=2}^{r} (R_{k_i}) \circ IW_{k_1}(M),$$

where R, IW and PW are a round routine, a prewhitening and a postwhitening routine, respectively. In Fig. 1 the key expansion is an algorithm that takes a master key K as input and produces the subkeys $k_1, k_2, \ldots, k_{r+1}$ for all stages of encryption.

A mixing key routine of a block cipher is an algorithm which injects a round key into an encryption routine. In the majority of modern block ciphers the mixing key function is implemented using the XOR operation because of its low-cost implementation.

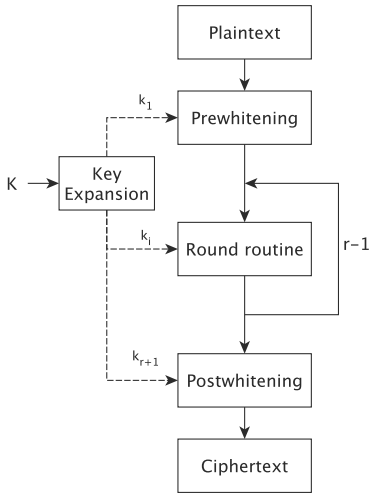

Fig. 1: *The general structure of an iterative block cipher*

To be an advanced algorithm, a modern block cipher should satisfy the following requirements [5]

- the complexity of the encryption and decryption has to be commensurate with the current standards;
- be protected against all known and prospective attacks;
- have high performance on widespread platforms.

It is quite challenging to satisfy the last point. Nevertheless, there are many publications regarding high performance implementations of AES. This is due to the fact that it is the most widespread block cipher and therefore the most optimized cryptographic algorithm for variety of platforms. However, getting into the Internet of things era, where devices communicate with each other via secure channels, it became necessary to have lightweight primitives. A lightweight cryptographic algorithm possesses a practical security level with enough performance in resource-limited settings: clock-cycles, area or energy [33].

1.2.2. AUTHENTICATED CIPHERS

There has been insufficient time to identify the generalized model of authenticated ciphers. There are only general structures such as encrypt-then-MAC, encrypt-and-MAC and MAC-then-encrypt [34]. Therefore, this section will focus on the general ideas and issues underlying these algorithms.

As mentioned earlier, in addition to confidentiality of transmitted information it is often required to ensure its integrity. Due to limitations of equipment and large amounts of processed data, the application of asymmetric cryptography for these purposes is not always possible. Therefore, an encrypted message is processed by a message authentication code to produce a tag [35]. Lots of modern MACs are based on block ciphers. These belong to the group of symmetric algorithms. Some of them are standardized and widely used in everyday life [36].

In general MACs only provide data integrity. Moreover, the complexities of the tag calculation and the message encryption are commensurable. In other words, to provide both confidentiality and integrity, two transformations of approximately equal complexity must be performed sequentially. In order to reduce the amount of transformations and system bandwidth, special algorithms have been developed [36]. The next generation authenticated cipher will be chosen after CAESAR.

Most authenticated schemes are nonce-based, i.e. an initialization vector (nonce) is transmitted together with data [37]. This solution helps to protect the algorithm against replay attacks and to use a pre-shared key for many messages. In addition, authenticated ciphers can operate in associated data mode [38]. This mode allows to encrypt only part of the data while the tag is generated for the entire message. This property is a useful addition in many situations where part of the message must be transmitted in plaintext. An Internet protocol (IP) packet is the most obvious example due to its widespread distribution. While the body of the packet can contain encrypted data, service information (e.g., data ports, IP addresses of sender and recipient, etc) has to be in clear to maximize data transfer speed.

From a security point of view the requirements imposed on authenticated ciphers include everything from block ciphers and message authentication codes [39]. Game theory is often used to prove the security of algorithms. However, all specific attacks applied to block ciphers and MACs can be easily adapted to authenticated ciphers (see Section 1.3). Security evaluation of

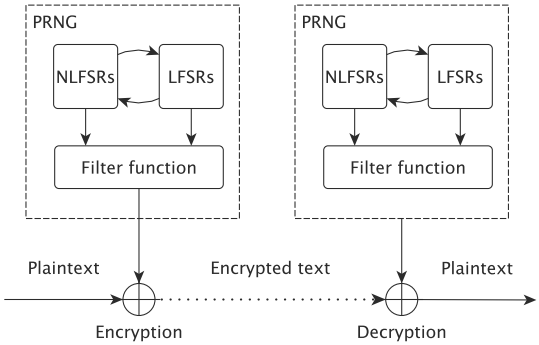

Fig. 2: *The overall structure of a stream cipher*

authenticated encryption algorithms therefore is more complex and consequently requires more resources.

1.2.3. STREAM CIPHERS

The main feature of stream ciphers is generation of random numbers (keystream) based on an initialization vector and key. Further, the plaintext is divided into chunks and added with the keystream using modulo operations to form the ciphertext. Since stream ciphers are typically targeted at hardware implementations, the XOR operation is often used instead of additional modulo [12, 40, 41].

Modern stream ciphers consist of linear and nonlinear feedback shift registers (LFSRs and NLFSRs), and a filter function to achieve maximum resistance against advanced attacks. Fig. 2 depicts the overall structure of a stream cipher.

In most cases, registers do not work independently, and operate in so-called mutual control mode. In other words, the states of the registers depend not only on their previous states, but also on other components of the cipher. If the keystream is generated randomly and without period, then an adversary cannot even theoretically recover the ciphertext [6, 42]. However, the practical application of such a scheme is too limited. Therefore, a key of fixed length is used to generate a pseudorandom sequence satisfying a number of criteria, including Golomb's randomness postulates [41].

Many designers of stream ciphers provide security proofs using a number of assumptions, which hypothetically could lead to vulnerabilities [12]. As a consequence the question regarding the theoretical proof of the security of NLFSRs-based ciphers remains open.

1.2.4. Cryptographic hash functions

A hash function is a method for mapping data of arbitrary size to a fixed-length value (hash code or hash value). Cryptographic hash functions are the subset of hash functions, which are resistant to at least 3 attacks: pre-image, second pre-image and collision [2, 6]. These criteria are classic and the most general, i.e. applicable for any cryptographic hash function. However, the practical application introduces it's own criteria for these cryptographic primitives. For example, performance and protection against all known attacks were the main criteria while selecting functions at the SHA-3 competition [23].

The existence of one-way hash functions has not been theoretically proven. It is assumed that the determination of the input message is a time consuming task. For example, the "birthday paradox" attack allows to find a collision after about $2^{\frac{n}{2}}$ calls of the hash function with an n-bit length hash code. Therefore, the hash function has resistance to the collision attack if and only if there is no algorithm with a complexity less than $2^{\frac{n}{2}}$ [2].

By default (sometimes used as a criterion) it is assumed that a slightest change (e.g., bit inverse) in the input message leads to significant changes in the hash value. This criterion is also known as the avalanche effect and plays a very important role when the hash function is used for generation of pseudorandom sequences [43].

Modern cryptographic hash functions have three main stages to compute the hash code (Fig. 3) [24]

- initialization based on IV (IS);
- partitioning the input message (M) into blocks and consistent application of a compression function (CF) to each of them;
- final transformations and generation of the output (FS).

Most modern hash functions were constructed using the Merkle-Damgård scheme [6, 44, 45]. In the last 10 years many undesirable features have been found in this approach, including the length extension attack [46]. During the

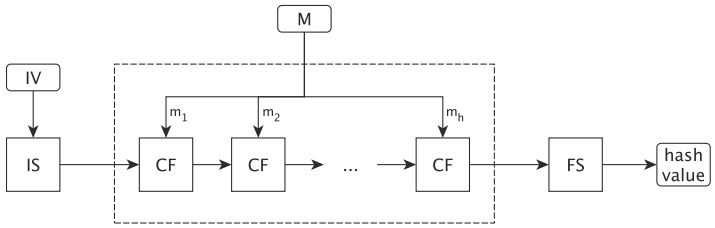

Fig. 3: *The high-level scheme of a hash function*

SHA-3 competition, a well-proven alternative construction called sponge was introduced [47]. It can be used to design authenticated and stream ciphers, message authentication codes, and other symmetric primitives. Moreover, this method of construction is the basis of the algorithm Keccak, which became the winner of SHA-3 [24].

1.3. Methods of cryptanalysis

1.3.1. Differential

Differential cryptanalysis implies the existence of ordered pairs (α, β) such that a randomly chosen plaintext M and the corresponding value $M - \alpha$ map to ciphertexts C and C', respectively [48]. Denote by $\beta = C - C'$ the difference between the ciphertexts, where "$-$" is the operation inverse to the mixing key routine. The ordered pair (α, β) is called the differential. The set of differentials at different rounds for a certain cipher is termed the differential characteristic [5, 48]. The attack is more effective for higher differential probability (at the same time not equal to 1). While the most general case is considered in [49, 50], in this section it is assumed that "$-$" is equivalent to XOR.

To apply the attack a difference distribution table is calculated for a given substitution. The maximum value of the table (MDT) excluding the value of the first row and first column, is calculated as follows [51]

$$\delta = \max_{\alpha \in \mathbb{F}_2^n, \alpha \neq 0, \beta \in \mathbb{F}_2^m} \#\{x \mid S(x) \oplus S(x \oplus \alpha) = \beta\},$$

where S is an S-box used in a cryptographic primitive.

During the differential attack an adversary learns how the difference of plaintexts affects the resulting difference (ciphertexts) [5]. The differential

propagated with the highest probability is used to find a round key. For the most modern block ciphers it is enough to break the entire encryption algorithm.

1.3.2. LINEAR

Linear cryptanalysis is based on the Piling-up lemma and was first applied to the block cipher FEAL [52]. Later Nyberg described the concept of the attack and Matsui has shown a practical example for the block cipher DES [53, 54]. The basic idea of linear cryptanalysis is based on the following statement. For randomly chosen bits of the key (k), plaintext (m) and ciphertext (c) the probability of the expression $\alpha \cdot m + \beta \cdot c + \gamma \cdot k$, where "$\cdot$" denotes the scalar product, differs from $\frac{1}{2}$ [5]. Let S be a substitution with n-bit input and m-bit output, and λ be the maximum value of an approximation table (excluding the value of the cell [0,0]) [51]. Then

$$\lambda = \max_{\alpha \in \mathbb{F}_2^n, \alpha \neq 0, \beta \in \mathbb{F}_2^m} \left| \#\left\{ x \mid \bigoplus_{s=0}^{n-1} (x_s \cdot \alpha_s) = \bigoplus_{t=0}^{m-1} (S[x]_t \cdot \beta_t) - 2^{n-1} \right\} \right|,$$

where γ_j is jth bit of γ. Linear cryptanalysis is more efficient for the greater value of λ [5].

1.3.3. ALGEBRAIC

Algebraic cryptanalysis exploits algebraic features of cryptographic algorithms. Whilst algebraic attacks on stream ciphers are well studied from both a theoretical and practical point of view [51, 55–57], for others cryptoprimitives the question remains open. In this connection, the following description will be based on known results for block ciphers. The same approach can be applied for other cryptographic primitives such as authenticated ciphers, hash function, etc.

During an algebraic attack the encryption algorithm is often represented as a system of equations. To obtain the key it is necessary to solve the system with respect to all variables. It is believed that the system with a lower degree is easier to solve [55]. To implement the attack, the following stages must be performed

- decompose the encryption algorithm into basic components;
- describe each of the elements algebraically;

- bind each of the output values to the input of other components.

Decomposition is a partition of the encryption algorithm into smaller pieces. By a basic component in modern ciphers linear and nonlinear transformations (layers) are understood [5]. An algebraic description is the conversion of the main elements into a system of equations that holds for all input and output values of the transformations. The output of these stages is the system of equations describing the entire encryption (decryption) algorithm including the key expansion routine.

To date there are many methods for solving systems of equations over $\mathbb{F}_2$ such as Gaussian elimination, XL, F4 and Gröbner basis [55, 57]. Moreover, the complexity of most methods depends on the density of the system. This allows one to conclude that the density of the system of equations describing the substitution affects the complexity of the final system.

This method was fist applied to block ciphers by Courtois in the early 2000s [58]. His approach is based on the principle stated above, that is the description of the substitution by the system of equations with the gradual expansion for the entire encryption algorithm. Application of this approach allows to describe AES with a system of equations of degree 2.

Later theoretical results of Courtois were implemented by Weinmann in practice [59]. He attacked a scale-down version of the AES cipher (MiniAES), and thus demonstrated the viability of the algebraic attack. Similar results were obtained by Kleiman in [60]. Unlike Weinmann, she presented a general algorithm based on a matrix approach for obtaining the system of equations describing a given S-box. However, to break 4 rounds of the 16-bit version of AES was not possible, even with enormous computing resources [60]. A few years ago it was announced that a special case of the Gröbner basis algorithm can break up to 10 rounds of scaled-down AES [61].

In 2006 Courtois demonstrated an attack on a full version of 6-round DES [62]. Only one plaintext/ciphertext pair was necessary to find a key (20 bits of which have been fixed) on a personal computer.

Application of the algebraic attack was also demonstrated for ciphers submitted to the Ukrainian competition [63, 64]. Many designers have used Nyberg's method, i.e. calculation of the inverse element in the field $\mathbb{F}_{2^n}$ followed by an affine transformation, to generate substitutions [65]. This approach allows to achieve the best known indicators for protection against differential and linear cryptanalysis. However like in AES, the entire cryptoprimitive can be described by a system of equations of degree 2 [55, 63]. This is an

undesirable property that may cause future attacks. As a consequence, the analysis showed that substitutions used in the ciphers Kalyna and Mukhomor had a number of advantages over other ciphers [21].

There are other trends in the solution of the system of equations such as conversion to the SAT problem [57]. Moreover, in Paper I an essentially different approach to the description of the cryptographic primitive is shown where the degree of the system does not affect the complexity.

1.3.4. Related-key

A related-key attack is a kind of cryptanalysis where an adversary can observe the input and output of a cipher under the influence of different keys. She only knows mathematical relations of the keys whilst the exact values are initially unknown [66].

During this attack it is assumed that the cryptanalyst has no direct access to the searched key (e.g., the key is stored in a hardware encryption unit). Nonetheless, the adversary can change in a certain way different pieces of the key. Due to these limitations, the related-key cryptanalysis is more theoretical than practical. Nevertheless, it allows one to find the key with the minimal known complexity [67].

It should be noted that one of the main components of the biclique attack on AES is a correlation of the round keys [68]. The biclique attack became widespread after the successful implementation on that cipher. The authors of [68] have theoretically proved that the encryption key can be found with a complexity less than exhaustive search.

1.3.5. Combination of the methods

Nowadays it is almost impossible to apply independent attacks against modern ciphers. This is due to the fact that the designers take into consideration all known attacks when a new cipher is created. Differential and linear cryptanalysis in the form which has been applied to DES is already ineffective against present-day ciphers. Thereby, modified or combined attacks have begun to develop rapidly.

Because of the simplicity of the description a lot of attacks based on the differential properties have been developed during the last 20 years. These include truncated differential, impossible differential, boomerang, higher order differential and others attacks [5, 10].

In 2011 the full version of the cipher GOST 28147 was firstly attacked by sequential application of fixed points, meet-in-the-middle and brute force attacks [17]. The same year the first attack on the full version of the cipher AES was published [68]. This attack consists of a combination of related-key and brute force attacks with the help of a complete bipartite graph.

Thus, the development and application of combined methods is a priority area of research in cryptology.

1.3.6. Other directions

It is assumed that even weak ciphers can become cryptographically strong when increasing the number of rounds. However, unlike the others, in slide attacks an adversary analyzes the key expansion rather than looks for vulnerabilities in the encryption routines [69]. This type of attack was firstly proposed by Wagner and Biryukov in [70]. It is mainly applied to iterative ciphers, a part of which (usually the round function) is applied sequentially by using only one key. The important thing in this attack is that the part must be identical and invertible. Thus, the number of cycles of the algorithm in this case does not affect the success of its breaking.

In recent years, the number of papers on cryptanalysis which do not consider the internal structure of the cipher is constantly increasing. For example, in [71, 72] it was shown that if an adversary has access to the session key management then she could restore a long-term key of the cipher GOST 28147 in a few minutes. Another example in this direction is Isobe's attack [73]. It is based on the ratio of the round key lengths to the block size while the round routine of the cipher is represented as a random function. More general theoretical results consist in finding distinguishers for universal schemes (Feistel, Lai-Massey, SPN and Sponge). The analysis shows advantages of one construction over another under the condition of the random or permutation round function. [74].

Side channel attacks should also be mentioned [75]. They use power or time fluctuations, leakage through electromagnetic or sound media, and other sources for obtaining information about the master key. Side channel attacks relate to attacks on implementation. Even theoretically secure encryption algorithms can be broken due to poor software or hardware implementation. However, practical experiments show that in some cases it is possible to create additional criteria to the basic components, thereby increasing the complexity of certain side channel attacks [76].

2. BINARY NONLINEAR MAPPINGS IN CRYPTOGRAPHY

Analysis of the latest solutions used in constructions of advanced cryptographic primitives allows to conclude that they largely inherited ideas of the block cipher AES [10]. Unlike Rijndael, where the substitution was generated based on Nyberg's design, new ciphers have one or more randomly generated S-boxes. Their main advantage is a description by a system of equations of degree 3 [77].

Substitutions for modern symmetric primitives are usually implemented in the form of lookup tables. Considering that lots of symmetric algorithms (e.g., Rijndael, PRESENT, ARIA, Keccak, etc.) use XOR as the key mixing routine, S-boxes are the only elements defining nonlinearity of encryption transformation and the level of resistance against cryptanalytic attacks [5]. Moreover, the number of encryption cycles is calculated based on cryptographic parameters of a nonlinear mapping, given in advance.

Aspects of vectorial Boolean functions used in symmetric cryptography as substitutions and their relevant cryptographic properties are presented in this section.

2.1. DEFINITIONS AND NOTATIONS

Let n and m be two positive integers. Define by $\mathbb{F}_2^n$ a vector space of all binary vectors of length n, where $\mathbb{F}_2$ is the Galois field with elements $\{0, 1\}$. Then an (n, m)-function is a vectorial Boolean function $F : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$. Boolean functions $f_1, f_2, \ldots, f_m$, such that $F(x_1, \ldots, x_n) = (f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n))$, and their linear combinations are called coordinate and component functions of F, respectively. If $m = 1$ then a vectorial Boolean function has a single output bit and is called a Boolean function. To find algebraic properties of (n, m)-functions, a vector space is often induced by a structure of the finite field $\mathbb{F}_{2^n}$.

For any positive integers n and m, a function F from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$ is called differentially δ-uniform if for every $a \in \mathbb{F}_2^n \setminus \{0\}$ and every $b \in \mathbb{F}_2^m$ the equation $F(x) + F(x + a) = b$ admits at most δ solutions [65, 78]. Vectorial Boolean functions used as S-boxes in cryptographic primitives must have low differential uniformity to provide high resistance to differential cryptanalysis (see Subsection 1.3.1). For the special case $n = m$ differentially 2-uniform functions are called almost perfect nonlinear (APN). Since $\delta \geq 2$, they are optimal regarding this criterion. The notion of APN function is closely related

to the notion of almost bent (AB) function [79]. The last one can be described in terms of the Walsh transform for a function $F : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$

$$\lambda(u, v) = \sum_{x \in \mathbb{F}_2^n} (-1)^{v \cdot F(x) + u \cdot x},$$

where $u \in \mathbb{F}_2^n$, $v \in \mathbb{F}_2^m$ and "$\cdot$" denotes scalar products in $\mathbb{F}_2^n$ and $\mathbb{F}_2^m$, respectively.

The set $\{\lambda(u, v) \mid (u, v) \in \mathbb{F}_2^n \times \mathbb{F}_2^m, v \neq 0\}$ is called the Walsh spectrum of F. If $n = m$ and the Walsh spectrum of F consists of $\{0, \pm 2^{\frac{n+1}{2}}\}$ then the function F is called AB [79]. AB functions exist for n odd only and oppose an optimal resistance to linear cryptanalysis (see Subsection 1.3.2). Every AB function is APN but the converse is not true in general (see [51, 80] for a comprehensive survey on APN and AB functions).

The natural way of representing $F : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ is algebraic normal form (ANF)

$$F(x_1, x_2, \ldots, x_n) = \sum_{I \in \mathbb{P}(\{1,\ldots,n\})} a_I \left(\prod_{i \in I} x_i \right), \qquad a_I \in \mathbb{F}_2^m,$$

where $\mathbb{P}(z)$ denotes the power set of z. The algebraic degree $deg(F)$ of F is the degree of its ANF. F is called affine if $deg(F)$ is at most 1. An affine vectorial Boolean function with $F(0, \ldots 0) = 0$ is linear.

2.2. Cryptographic properties of vectorial Boolean functions

While Boolean functions are adopted mainly as filtering functions in stream ciphers, vectorial Boolean function are used in block and authenticated ciphers, and hash functions as substitutions. For theoretical analysis the univariate representation is one of the best ways to consider cryptographic properties of the binary mappings. However, field operations are not so well optimized as operations with Boolean functions in modern computers, especially for large n. Therefore, it makes sense to represent cryptographic properties of (n, m)-functions using the set of component functions. All definitions and indicators are well-known and one can see [51, 80] for more details.

First of all, let's consider the properties of Boolean functions. A Boolean function of n variables is called balanced if $\sum_{x=0}^{2^n-1} f(x) = 2^{n-1}$, where $x =$

$(x_1, x_2, \ldots, x_n)$. The correlation between an arbitrary Boolean function $f(x)$ and the set of all linear functions is determined by Walsh transformation

$$W(w) = \sum_{x=0}^{2^n-1} (-1)^{f(x) \oplus l_w(x)},$$

where $l_w(x) = w \cdot x = w_1 x_1 \oplus w_2 x_2 \oplus \ldots w_n x_n$. The nonlinearity is related to the Walsh values as

$$NL(f) = \frac{1}{2} \left(2^n - \max_{w \in \mathbb{F}_2^n \setminus \{0\}} |W(w)| \right).$$

Autocorrelation of f noted as $r_f(\alpha)$ shows how the function differs from itself shifted on several positions, i.e.

$$r_f(\alpha) = \sum_{x=0}^{2^n-1} (-1)^{f(x) \oplus f(x \oplus \alpha)},$$

where $\alpha \in \mathbb{F}_2^n$. For cryptography the maximal value of the function $r_f(\alpha)$ is of interest, and can be found as

$$AC_{max}(f) = \max_{\alpha \in \mathbb{F}_2^n \setminus \{0\}} \left| r_f(\alpha) \right|.$$

Let σ be the sum-of-squares indicator, then

$$\sigma = \sum_{\alpha=0}^{2^n-1} r_f^2(\alpha).$$

Let $hw(\alpha)$ be a binary Hamming weight of $\alpha \in \mathbb{F}_2^n$ [51]. Then it is said that $f(x)$ satisfies propagation criterion of order k ($PC(k)$) if and only if for all nonzero vectors $\alpha \in \mathbb{F}_2^n$ such that $1 \leq hw(\alpha) \leq k$ the following is true

$$\sum_{x=0}^{2^n-1} f(x) \oplus f(x \oplus \alpha) = 2^{n-1}.$$

The strict avalanche criterion (SAC) corresponds to $PC(1)$.

A Boolean function is correlation immune of order m ($CI(m)$) if the equation $W(w) = 0$ holds for all $w \in \mathbb{F}_2^n$, where $1 \leq hw(w) \leq m$. If the function is

balanced and satisfies $CI(m)$ simultaneously, then such a function is called m-resilient.

The minimum algebraic degree of $g(x) \neq 0$ of the set $\{g \mid f(x) \cdot g(x) = 0\}$ $\cup \{g \mid (f(x) \oplus 1) \cdot g(x) = 0\}$ is called algebraic immunity (AI) of a Boolean function f.

Using the above definitions let's describe cryptographic properties of substitutions. Suppose S is the table representation of a vectorial Boolean function $F = (f_1, \ldots, f_m)$ from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$. Define $\{h_j = j \cdot F \mid 0 < j < 2^m\}$ as the set of the component functions of F. Then

- nonlinearity of S is
$$NL(S) = \min_{0<j<2^m} \left(NL(h_j)\right);$$
- minimum degree of S is
$$deg(S) = \min_{0<j<2^m} \left(deg(h_j)\right);$$
- the maximum value of autocorrelation spectrum of S is
$$AC_{max}(S) = \max_{0<j<2^m} \left(AC_{max}(h_j)\right);$$
- S satisfies strict avalanche criterion if every h_j satisfies SAC;
- S satisfies propagation criterion of order k if every h_j satisfies $PC(k)$;
- S is correlation immune of order k if every h_j is $CI(k)$;
- S is balanced (permutation) if every h_j is balanced;
- S is k-resilient if every h_j is k-resilient.

Similar properties for vectorial Boolean functions are given in [51].

While the maximum value of the approximation table (λ) can be calculated directly from the nonlinearity of the S-box as $\lambda = 2^{n-1} - NL(S)$, the maximum value of the differential table cannot be directly evaluated from the component functions. For the given S-box the indicator δ-uniformity defined in 1.3.1 and 2.1 is equivalent to the maximum value of MDT.

The ways to represent a substitution as a system of equations over $\mathbb{F}_2$ are given in [60, 63]. Define density as the fraction of nonzero elements in a system of equations. Then, a substitution provides better protection against algebraic attacks (see 1.3.3) if the system

- has higher degree;
- has fewer equations;
- is more dense.

Unambiguous theoretical relations between these parameters is an unsolved problem [81]. Suppose the degree of a system of equations is the maximal algebraic degree of all polynomials this system consists of. Then algebraic immunity of the S-box ($AI(S)$) means the smallest degree of the system describing this substitution.

2.3. Equivalence of vectorial Boolean functions

Two functions $F, G : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ are called extended affine equivalent if there exist such affine permutations $A_1 = L_1(x) + c_1, A_2 = L_2(x) + c_2$ and an arbitrary linear function $L_3(x)$ such that

$$F(x) = A_1 \circ G \circ A_2(x) + L_3(x).$$

If $L_3(x) = const$, or $L_3(x) = 0, c_1 = 0$, and $c_2 = 0$ then F and G are affine, or linear equivalent, respectively. Moreover, for at least one missing element of $L_1(x), L_2(x), L_3(x), c_1, c_2$ the functions are called restricted EA (REA) equivalent [82].

Any affine function $A : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ can be represented in matrix form

$$A(x) = K \cdot x \oplus C,$$

where K is an $m \times n$ matrix and $C \in \mathbb{F}_2^m$. All operations are performed in $\mathbb{F}_2$, thus the above equation can be rewritten as

$$\begin{pmatrix} a_0 \\ a_1 \\ \cdots \\ a_{m-1} \end{pmatrix}_x = \begin{pmatrix} k_{0,0} & \cdots & k_{0,n-1} \\ k_{1,0} & \cdots & k_{1,n-1} \\ \vdots & \ddots & \vdots \\ k_{m-1,0} & \cdots & k_{m-1,n-1} \end{pmatrix} \cdot \begin{pmatrix} x_0 \\ x_1 \\ \cdots \\ x_{n-1} \end{pmatrix} \oplus \begin{pmatrix} c_0 \\ c_1 \\ \cdots \\ c_{m-1} \end{pmatrix}$$

where $a_i, c_i, x_s, k_{j,s} \in \mathbb{F}_2$. This representation allows to describe EA-equivalence in matrix form

$$F(x) = M_1 \cdot G(M_2 \cdot x \oplus V_2) \oplus M_3 \cdot x \oplus V_1$$

where elements of $\{M_1, M_2, M_3, V_1, V_2\}$ have dimensions $\{m \times m, n \times n, m \times n, m, n\}$.

In [83] F and G are considered as $G_F(x, y) = \{\{x, y\} \mid y = F(x)\}$. They are Carlet-Charpin-Zinoviev (CCZ) equivalent, if for $F_2(x) = L_3(x) + L_4 \circ G(x)$ and permutation $F_1(x) = L_1(x) + L_2 \circ G(x)$ the following equation holds

$$F(x) = F_2 \circ F_1^{-1}(x),$$

where $L_1(x)$, $L_2(x)$, $L_3(x)$, $L_4(x)$ are arbitrary affine functions.

CCZ-equivalence is the most general known equivalence of functions for which differential uniformity and extended Walsh spectrum are invariants. In particular every function CCZ-equivalent to an APN (respectively, AB) function is also APN (respectively, AB). EA-equivalence is a special case of CCZ-equivalence [51]. The algebraic degree of a vectorial Boolean function is invariant under EA-equivalence but, in general, it is not preserved by CCZ-equivalence.

3. Summary of Papers

This thesis is based on seven papers. A synopsis of each paper is given in the following subsections.

3.1. Paper I

Several approaches which use binary decision diagrams for algebraic attacks are well-known in open literature. The efficiency of BDD-based attacks is demonstrated both for general models and for particular cases such as A5/1, E0 and Trivium [84, 85]. In this paper we extend the previous results on block ciphers and present new specific strategies and approaches for solving compressed right hand side (CRHS) systems [86].

Most ciphers use only one nonlinear element, which is usually represented as a lookup table. Hence, we are interested in finding a BDD that represents a given S-box mapping $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$. Let the input and output bits of the S-box be $x_0, \ldots, x_{n-1}$ and $y_0, \ldots, y_{m-1}$, respectively. Denote the levels of a binary tree as $\{x_0, \ldots, x_{n-1}, y_0, \ldots, y_{m-1}\}$. For each value of substitution create a path from the source node on top to the sink node (true node) at the bottom, and all edges direct downwards. If the edges are divided into 0-edges and 1-edges, then we can uniquely represent an arbitrary S-box using a BDD upto the order of variables.

Since each level is represented in general as a linear combination of input and output bits, then a linear layer of a cryptographic primitive does not add extra variables and as a consequence does not affect the complexity of the operations performed on the tree [86]. This representation allows to join several BDDs using adjacent variables on different levels. Thereby, the entire encryption algorithm can be described as one big BDD or as the set of smaller BDDs.

Several operations such as swapping and adding levels, and absorbing linear dependencies are also defined on this special version of BDD. While joining together many BDDs and absorbing all linear dependencies, the solving complexity depends heavily on the order the BDDs are joined. Finding the ordering of BDDs that gives the minimum complexity is probably a hard problem. During our experiments we have not found a strategy for ordering that is universally best. However, we described automatic ordering, divide-and-conquer and order by cryptanalysis strategies for how to join and absorb, with the aim to keep the complexity down.

We apply the proposed attack on DES with a reduced number of rounds, MiniAES and the EA-equivalence problem. Our experiments have shown that 6-round DES can be broken in approximately one minute on an ordinary computer. This is a factor 2^{20} improvement over the best earlier algebraic attack on DES using MiniSAT [62].

There have been several earlier attempts to break MiniAES [60, 61, 87]. Approaches that exploit the short key in MiniAES (only 16 bits) succeed very quickly, but the general methods of F4 and XL/XSL failed to solve systems representing more than one round of MiniAES. The approach we use in the paper does not exploit the short key, while still solving systems representing 10 rounds of MiniAES using approximately 45 minutes and 8GB of memory. In addition, the BDD method has shown the advantages compared to a Gröbner basis and CryptoMiniSat for solving the EA-equivalence problem.

Despite the excellent practical results, a number of unresolved issues still remain. The main one concerns the theoretical estimates of the complexity of the BDD attack.

3.2. PAPER II

For most new algorithms evaluation of the resistance to known attacks, such as differential, linear or algebraic, is provided by the designers. However, an independent verification of the results is always needed [88]. To conduct such

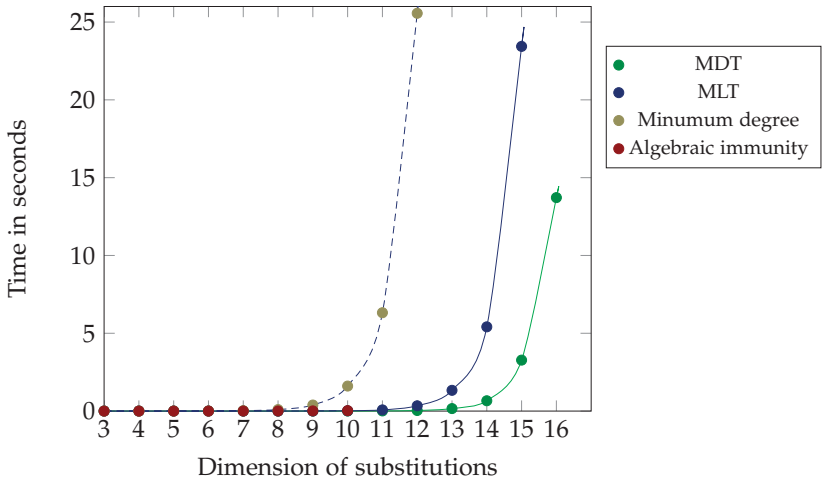

Fig. 4: *The relationship between the dimension of random substitutions and time of calculation*

research, tools for analysis of both basic components and entire encryption algorithms are required. On the other hand, universal approaches would also be a useful supplement for the designers of prospective algorithms. Choosing linear layers is a relatively simple task when only few indicators are considered [89]. The situation is completely opposite for nonlinear layers which usually consist of parallel application of substitutions.

As was mentioned in Section 2 vectorial Boolean functions have lots of cryptographic properties. While for a given S-box some properties are calculated directly from the formula, others require special knowledge (i.e. for algebraic immunity). Today there are a number of tools that can be considered a partial solution to the problem [57, 90–93]. However, the cryptographic community needs a universal approach to calculate indicators for arbitrary binary mappings. In this paper a tool for generating and analyzing arbitrary vectorial Boolean functions $F : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ was given.

The proposed library (package) S-box includes methods for calculation of all indicators described in Section 2. In particular one can find δ-uniformity, nonlinearity or maximum of the linear approximation table, minimum degree, algebraic immunity, maximum value of autocorrelation spectrum, correlation immunity and other cryptographic properties for arbitrary vectorial Boolean functions. In addition to this, there are implemented several methods for

Table 1: *Comparison of 8-bit S-boxes*

Properties	AES	GOST R 34.11-2012	STB 34.101.31-2011	Kalyna's S0 [21]	Proposed in [77]
MDT	4	8	8	8	8
NL	112	100	102	96	104
Absolute indicator	32	96	80	88	80
SSI	133120	258688	232960	244480	194944
Minimum degree	7	7	6	7	7
Algebraic immunity	2	3	3	3	3

generating substitutions with predefined properties based on Gold, Kasami, Welch, inverse and other well-known functions. The library also contains a number of auxiliary functions such as finding the univariate polynomial or the system of equations describing the substitution; checking the APN property, or CCZ- equivalence; generating look up tables based on the user-defined univariate polynomials and many others.

The performance and arbitrary dimension of binary nonlinear mappings were the main criteria for the S-box library. Calculation of some indicators are based on known results [94, 95], while others (i.e. cyclic properties or algebraic immunity) were optimized during the research and experiments. Fig. 4 shows the time complexity of several frequently used methods for $n = m$.

From a practical point of view, Sbox can be used to analyze nonlinear components of the existing or prospective cryptographic primitives. An example of the substitution comparison is given in Table 1.

In conclusion, the library includes lots of functions for computing the properties of permutations and methods of generation. Despite this, there are many directions for improvement and development. The library is designed to facilitate extension of its functionality quite easily, for instance by combining to optimize methods for calculation of indicators such as minimum degree or autocorrelation, or by realizing a native integration with Sage and creation of a universal test environment.

3.3. PAPER III

Since substitutions are one of the main components that determine the security of modern cryptographic algorithms, many cryptographic criteria must be considered for a new cryptographic primitive. Taking into account the large number of existing indicators, their controversy and partial interdependence, it is most likely impossible to generate a substitution that satisfies all known requirements. This became a reason to use a substitution satisfying only mandatory criteria essential for a particular symmetric algorithm. Such substitutions are called optimal [10, 51, 80]. Optimality criteria may vary depending on which cipher is considered.

After investigation of existing and prospective attacks the following criteria were highlighted as significant

- maximum value of minimum degree;
- maximum algebraic immunity with the minimum number of equations;
- absence of fixed points (cycles of length 1);
- substitution must be bijective (permutation);
- minimum value of δ-uniformity and maximum value of nonlinearity limited by parameters listed above.

In particular, for $n = 8$ an optimal permutation has algebraic degree 7, algebraic immunity 3 and 441 equations, δ-uniformity under 8, nonlinearity over 100 and without fixed points.

The majority of theoretical methods for generation of vectorial Boolean functions have extreme characteristics of δ-uniformity and nonlinearity, but at the same time do not possess other properties (i.e., high value of algebraic immunity) which are necessary for next-generation symmetric cryptographic primitives.

The first and most obvious solution is to generate random permutations and check them on optimality. After 12 hours of cluster operation (4096 cores) there were found 27 optimal permutations with $NL = 100$. Four of them were CCZ-inequivalent. After 48 hours (22 years on 1 core) the program run on the same cluster didn't find any substitution with $NL \geq 102$.

A counterexample was found in STB 34.101.31-2011 [20]. The optimal substitution has $NL = 102$. Thus we found another way to generate substitutions with $NL \geq 100$. Instead of trying to find a random permutation or apply the

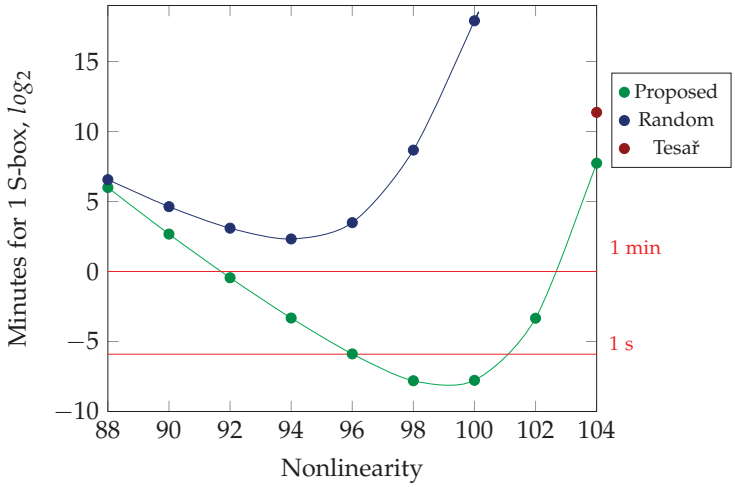

Fig. 5: *Performance comparison of the substitution generation methods*

hill climbing technique, it was decided to solve the problem from the other side. We started with the best known permutation and modified it (swapped values) until the expected result was achieved. As was proven in [96] one swap does not much influence nonlinearity and δ-uniformity.

Before presenting our result at the conference we found another algorithm which produces the same result [97]. The performance comparison (Fig. 5) shows that our proposed method is 10 times faster then Tesař's [97].

After 107 hours of cluster operations, that are equivalent to 50 years on a single-processor computer, there were not found better substitutions. The practical results of both methods show that there are no optimal substitutions with nonlinearity greater than 104. However, there are permutations with nonlinearity 106 and algebraic immunity 2, in which the number of equations is small (e.g. 1). Hereby, the question about existence of optimal substitutions with nonlinearity more than 104 remains open.

Four substitutions used in the new Ukrainian block cipher and hash function were generated by the proposed method. An additional criterion which the substitutions must satisfy is belonging to different CCZ-equivalent classes. Details stated in Paper V.

3.4. PAPER IV

In 2010 at the RusCrypto'10 conference a prototype of the prospective hash function also known as Stribog (Steebog) [27, 98] was presented. Two years latter this hash function was accepted as the governmental standard GOST R 34.11-2012 [29]. The description of the hash function available in public literature was only algorithmic. To prove some cryptographic properties it is necessary to have a common mathematical representation as has been made in this paper.

The core of the hashing algorithm is the $L \circ P \circ S$ transformation. Transformation of the state into an 8×8 byte matrix gave a general idea of each transformation. Further investigations showed that S and P transformations have analogues in AES. While S is identical to the SubBytes routine, P is similar to ShiftRows. Unlike ShiftRows, P transposes the state instead of shifting it by a constant number positions. The most difficult task was to identify the L transformation, which is a multiplication by a 64×64 binary matrix.

In summary, the main issue was to find the irreducible polynomial which gives the representations of transformations over $\mathbb{F}_{2^8}$ that produce the same outputs. Based on the assumption that the matrix used in L possesses the MDS property, the polynomial $f(x) = x^8 + x^6 + x^5 + x^4 + 1$ was found. Using this polynomial all basic components of the hash function were described in AES-like form.

This representation allows to use the wide trail strategy to prove the resistance of the hash function to differential and linear cryptanalysis. At the same time, the existing attacks can be easily adapted to GOST R 34.11-2012 [99]. Additionally, this gives access to well-known optimization techniques for increasing performance on a variety of platforms [10]. Using a table approach a fast cross-platform implementation of Stribog was proposed [100, 101].

3.5. PAPER V

As stated before, the choosing of essential properties for new substitutions is not a trivial task. In this paper an analysis of the absence of fixed points criterion is given. If one considers the round function instead of a single substitution, then even for the AES S-box fixed points can be achieved.

The investigation is based on the fact that a cipher has lots of isomorphic (equivalent) representations. For AES the ShiftRows, MixColumns and

AddRoundKey routines are linear transformations with respect to XOR. Manipulations with these transformations give different representations [55]. It is shown that at least one fixed point can be found for the AES substitution in case of using XOR operation in AddRoundKey.

Applying the same model the advantages of additional modulo 2^n were shown. A mixing key routine based on the modulo operation adds more nonlinearity and as a result reduces the number of possibilities for adversaries. The analysis shows the necessity of additional requirements for multiple substitutions used in one cryptographic primitive.

Proposition 1. *Substitutions S_1, S_2, ..., S_l used in a nonlinear layer must belong to different classes of equivalence.*

Since CCZ-equivalence is the most general case of known equivalences, it makes sense to check whether substitutions belong to different CCZ-equivalence classes.

The more practical result was achieved independently for Zorro [102]. The core of that attack has the same principles that were described in this paper.

3.6. PAPER VI

The behavior of nonlinear feedback shift registers is poorly understood, which, in turn, results in a lack of criteria for selecting parameters that directly affect security. To achieve this, designers of stream ciphers often combine linear and nonlinear registers. MICKEY is an example of such ciphers.

In several papers the theoretical weaknesses of MICKEY were presented [103–106]. It was shown in particular that choosing constants in the wrong way may lead to security problems. The shared idea of all these attacks is the construction of a backward states tree. After collecting the results from all previous papers it became possible to evaluate theoretically the probabilities of all possible branches in the tree. We proved both theoretically and practically that in key/IV load mode the expectation value of degree approximately equals 2. The analogous value for preclock and key-generation mode is approximately equal to 1. Thus, knowing the internal state of registers it is always possible to perform reverse steps to acquire the state after key initialization function. However, the inverse key/IV load modes produce a complete binary tree.

The other parts of the paper describe some practical observations. First, it is noted that each reverse step increases the probability of subtree cutting off

Table 2: *Complexities for Solving REA-equivalence Problem*

#	Restricted EA-equivalence	Complexity	$G(x)$
1	$F(x) = M_1 \cdot G(M_2 \cdot x)$	$O\left(n^2 \cdot 2^n\right)$	P
2	$F(x) = M_1 \cdot G(M_2 \cdot x \oplus V_2) \oplus V_1$	$O\left(n \cdot 2^{2n}\right)$	P
3	$F(x) = M_1 \cdot G(x \oplus V_2) \oplus V_1$	$O\left(2^{2n+1}\right)$	†
4	$F(x) = M_1 \cdot G(x \oplus V_2) \oplus V_1$	$O\left(n \cdot 2^{3n}\right)$	A
5	$F(x) = G(M_2 \cdot x \oplus V_2) \oplus V_1$	$O\left(n \cdot 2^n\right)$	P
6	$F(x) = G(x \oplus V_2) \oplus M_3 \cdot x \oplus V_1$	$O\left(n \cdot 2^n\right)$	A
7	$F(x) = M_1 \cdot G(x \oplus V_2) \oplus M_3 \cdot x \oplus V_1$	$O\left(2^{2n+1}\right)$	‡
8	$F(x) = M_1 \cdot G(x \oplus V_2) \oplus M_3 \cdot x \oplus V_1$	$O\left(n \cdot 2^{3n}\right)$	A

P - permutation; A - arbitrary;
† - G is under condition $\{2^i \mid 0 \leq i \leq m-1\} \subset \mathrm{img}(G')$ where $G'(x) = G(x) + G(0)$;
‡ - G is under condition $\{2^i \mid 0 \leq i \leq m-1\} \subset \mathrm{img}(G')$ where $G'(x) = G(x) \oplus L_G(x) \oplus G(0)$.

with all previous states. This property exists since there is a high probability of orphan states. Therefore, in some cases key bits could be found uniquely. Second, since the functions used for different modes are the same, it allows to generate key-streams shifted by a fixed number of bits for different pairs of key and IV. However, the conditions imposed on the use of keys and IVs stated in the MICKEY's specification do not give the opportunity to apply the attack in the real world. In the end, the meet-in-the-middle attack based on the backward states tree is proposed.

Taking into consideration everything mentioned above, the proposed method for analysis of MICKEY-like ciphers allows to justify the choice of the encryption algorithm parameters based on the estimation of branch points degree probabilities.

3.7. Paper VII

In [91] Alex Biryukov et al. have shown that in the case when given functions are permutations of $\mathbb{F}_2^n$, the complexity of determining their linear and affine equivalence equals $O\left(n^2 \cdot 2^n\right)$ and $O\left(n \cdot 2^{2n}\right)$, respectively. In Paper VII we

Table 3: *Practical Comparison of Solving REA-equivalence Problem*

#	n=6		n=8		n=10		n=12		n=14	
	ES	KM	ES	KM	ES	KM	ES	KM	ES	KM
1	69	12	125	14	197	17	285	20	389	22
2	81	15	141	19	217	24	309	28	417	32
3	47	13	79	17	199	21	167	25	223	29
4		21		27		34		40		46
5	47	12	79	14	199	17	167	20	223	22
6	48	9	80	11	120	14	168	16	224	18
7	83	13	143	17	219	21	311	25	419	29
8		21		27		34		40		46

consider the conditions under which the complexities of checking vectorial Boolean functions $F, G : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ on REA-equivalence can be reduced.

Matrix form is used for EA-equivalence representation in both presented and [91] methods. This approach allows to prove a number of propositions. Most of them are summarized in Table 2. The first two rows present the complexities from [91].

Proposition 2. *Any linear function* $L : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ *can be converted to a matrix with complexity* $O(n)$.

Since the considered functions have different REA-equivalent representations, the complexities can not be directly compared to each other. Therefore, Table 3 presents the comparison of known methods (KM) with exhaustive search (ES) based on the calculated complexities (in binary logarithm form) for most interesting values of n.

It is easy to see that for some of the above cases the complexity takes polynomial time. Obtained results give a practical method for checking arbitrary vectorial Boolean functions on REA-equivalence.

4. Conclusions

The research conducted solved a number of current important scientific tasks related to improving methods of cryptanalysis and developing of new requirements for advanced symmetric cryptoalgorithms. In particular, backwards

states cryptanalysis of the stream cipher MICKEY, and BDD-based algebraic attacks on DES and MiniAES show that even well-studied ciphers may have weaknesses. Consideration of these attacks at the design stage of new primitives enables to create better and more secure cryptographic algorithms.

In the post-AES era many cryptoprimitives providing high-level security have random substitutions. The main filtering criteria are balancedness, absence of fixed points, δ-uniformity, minimum degree, algebraic immunity and nonlinearity. At the same time, promising algebraic cryptanalysis is not yet fully understood, and the boundaries of its application are not clear.

A new heuristic method for generating S-boxes has been proposed based on the gradient descent method for generation of Boolean functions. It allows to generate substitutions with the best properties known to date at low cost resources. In particular, for $n = 8$ case the application of the method gives permutations with absence of fixed points, and indicators δ-uniformity 8, nonlinearity 104, minimum degree 7 and algebraic immunity 3. These substitutions surpass analogues used in standards STB 34.101.31-2011, GOST R 34.11-2012 and in the draft standard of the new Russian block cipher.

Advanced design approaches of symmetric cryptographic algorithms introduce additional requirements for S-boxes. One such requirement is that all permutations used in a nonlinear layer belong to different equivalence classes. Satisfying this reduces the number of weak isomorphic representations of an encryption algorithm. As a consequence, it becomes necessary to find equivalent transformations that can be used to construct isomorphic representations.

Several new methods for checking the equivalence of two binary nonlinear mappings have been proposed. These methods are based on the conversion of a linear function defined over a field $\mathbb{F}_{2^n}$ to the matrix form. Under certain conditions the complexity can be reduced to polynomial. The approaches used in proving of the proposed methods can be additionally applied to find original high-level representations of cryptographic primitives such as GOST R 34.11-2012.

The main practical result is the designed software for effective generation and calculation of indicators of arbitrary nonlinear binary mappings. This allows one to create and analyze arbitrary nonlinear components used in symmetric cryptographic primitives. Besides this, a patch for OpenSSL based on a cross-platform implementation of GOST R 34.11-2012 noted in Paper IV was created by Dmitry Olshansky [107]. Most of these results have also been

used in one of the Ukraine's leading companies that provides services in the field of information security.

REFERENCES

[1] SHIMEALL, T., SPRING, J.: *Introduction to information security: A strategic-based approach*. Syngress Publishing, 2013.

[2] SCHNEIER, B.: *Applied cryptography: Protocols, algorithms, and source code in C*. Wiley, 1996.

[3] GORBENKO, I., GORBENKO, Y.: *Applied cryptology: Theory. Practice. Application.* LLC Publishing "Fort", 2013. (In Ukrainian).

[4] VAN TILBORG, H. C., JAJODIA, S.: *Encyclopedia of cryptography and security*. Springer, 2011.

[5] KNUDSEN, L. R., ROBSHAW, M.: *The block cipher companion*. Information Security and Cryptography. Springer Berlin Heidelberg, 2011.

[6] MENEZES, A. J., VAN OORSCHOT, P. C., VANSTONE, S. A.: *Handbook of applied cryptography*. CRC Press, 2010.

[7] PRENEEL, B., BIRYUKOV, A., CANNIÈRE, C. D., ET AL.: NESSIE: Final report of European project number IST-1999-12324, named New European Schemes for Signatures, Integrity, and Encryption. *Electronic source,* 2004. `https://www.cosic.esat.kuleuven.be/nessie/Bookv015.pdf`.

[8] FIPS 46–3: Data Encryption Standard (DES). *National Institute of Standards and Technology,* 1993.

[9] LOUKIDES, M., GILMORE, J.: *Cracking DES: Secrets of encryption research, wiretap politics and chip design*. O'Reilly Media, 1998.

[10] DAEMEN, J., RIJMEN, V.: *The design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.

[11] FIPS PUB 197: Advanced Encryption Standard (AES). *National Institute of Standards and Technology,* 2001.

[12] ROBSHAW, M. J. B., BILLET, O. (eds.): *New stream cipher designs - The eSTREAM finalists*, vol. 4986 of *Lecture Notes in Computer Science*. Springer, 2008.

[13] CID, C., ROBSHAW, M., ET AL.: The eSTREAM portfolio in 2012. *Electronic source,* 2012. `http://www.cspforum.eu/D.SYM.10-v1.pdf`.

[14] CRYPTREC: Report of the cryptographic technique monitoring subcommittee. *Electronic source,* 2003. `http://www.cryptrec.go.jp/report/c03_wat_final.pdf`.

[15] CRYPTREC: Specifications of e-Government recommended ciphers. *Electronic source,* 2013. `http://www.cryptrec.go.jp/english/method.html`.

[16] Charnes, C., O'Connor, L., Pieprzyk, J., Safavi-Naini, R., Zheng, Y.: Comments on Soviet encryption algorithm. In De Santis, A. (ed.), *Advances in Cryptology — EUROCRYPT'94*, vol. 950 of *Lecture Notes in Computer Science*, pp. 433–438. Springer Berlin Heidelberg, 1995.

[17] Isobe, T.: A single-key attack on the full GOST block cipher. In Joux, A. (ed.), *Fast Software Encryption*, vol. 6733 of *Lecture Notes in Computer Science*, pp. 290–305. Springer Berlin Heidelberg, 2011.

[18] Dinur, I., Dunkelman, O., Shamir, A.: Improved attacks on full GOST. In Canteaut, A. (ed.), *Fast Software Encryption*, vol. 7549 of *Lecture Notes in Computer Science*, pp. 9–28. Springer Berlin Heidelberg, 2012.

[19] Aleksev, E., Smyshlyaev, S.: GOST 28147-89: "Do not rush to bury him." Part 1. Security of the algorithm. *Electronic source,* 2013. http://www.cryptopro.ru/blog/2013/08/27/gost-28147-89-ne-speshi-ego-khoronit-chast-1-stoikost-algoritma. (In Russian).

[20] STB 34.101.31-2011: Information technology and security. Information security. Cryptographic encryption algorithms and control of integrity. p. 35, 2011.

[21] Oliynykov, R., Gorbenko, I., Dolgov, V., Ruzhentsev, V.: Results of Ukrainian national public cryptographic competition. In *Tatra Mountains Mathematical Publications*, vol. 47, pp. 99–113. Mathematical Institute of Slovak Academy of Sciences, 2010.

[22] Oliynykov, R.: Next generation of block ciphers providing high-level security. *Winter School in Information Security, Finse,* 2014. `https://www.frisc.no/wp-content/uploads/2014/05/finse2014-oliynykov.pdf`.

[23] Kayser, R. F.: Announcing request for candidate algorithm nominations for a new cryptographic hash algorithm (SHA-3) family. In *Federal*

Register, vol. 72, pp. 62 212–62 220. National Institute of Standards and Technology, 2007.

[24] Chang, S.-j., Perlner, R., Burr, W. E., et al.: Third-round report of the SHA-3 cryptographic hash algorithm competition. *National Institute of Standards and Technology,* 2012.

[25] FIPS PUB 202 (draft): SHA-3 standard: Permutation-based hash and extendable-output functions. *National Institute of Standards and Technology,* May 2014.

[26] Grebnev, S., Dmukh, A., Dygin, D., Matyukhin, D., Rudskoy, V., Shishkin, V.: Asymmetric reply to SHA-3: Russian hash function draft standard. *In Pre-proceedings of Workshop on Current Trends in Cryptology (CTCrypt 2012),* 2012.

[27] GOST R 34.11-20__ (draft) revision 1: Information technology. Cryptographic data security. Hash function. *Electronic source,* 2010. `http://infotecs.ru/laws/gost/proj/gost3411.pdf`. (In Russian).

[28] Kazymyrov, O., Kazymyrova, V.: Algebraic aspects of the Russian hash standard GOST R 34.11-2012. *In Pre-proceedings of 2nd Workshop on Current Trends in Cryptology (CTCrypt 2013),* pp. 160–176, 2013.

[29] GOST R 34.11-2012: Information technology. Cryptographic data security. Hash-function. *Federal Agency on Technical Regulation and Metrology,* p. 34, 2013.

[30] Shishkin, V., Dygin, D., Lavrikov, I., Marshalko, G., Rudskoy, V., Trifonov, D.: Low-weight and hi-end: draft russian encryption standard. *In Pre-proceedings of 3rd Workshop on Current Trends in Cryptology (CTCrypt 2014),* 2014.

[31] Gauravaram, P., Knudsen, L. R., Matusiewicz, K., Mendel, F., Rechberger, C., Schläffer, M., Thomsen, S. S.: Grøstl – a SHA-3 candidate. *Submission to NIST (Round 3),* 2011. `http://www.groestl.info/Groestl.pdf`.

[32] CAESAR: Call for submissions, final (2014.01.27). *Electronic source,* 2014. `http://competitions.cr.yp.to/caesar-call.html`.

[33] Poschmann, A. Y.: *Lightweight cryptography: Cryptographic engineering for a pervasive world*. Ph.D. thesis, Ruhr University Bochum, Germany,

2009.

[34] Bellare, M., Namprempre, C.: Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In Okamoto, T. (ed.), *Advances in Cryptology — ASIACRYPT 2000*, vol. 1976 of *Lecture Notes in Computer Science*, pp. 531–545. Springer Berlin Heidelberg, 2000.

[35] Black, J., Halevi, S., Krawczyk, H., Krovetz, T., Rogaway, P.: UMAC: Fast and secure message authentication. In Wiener, M. (ed.), *Advances in Cryptology — CRYPTO' 99*, vol. 1666 of *Lecture Notes in Computer Science*, pp. 216–233. Springer Berlin Heidelberg, 1999.

[36] ISO/IEC 9797-1: Information technology – Security techniques – Message authentication codes (MACs) – Part 1: Mechanisms using a block cipher. *ISO/IEC JTC 1/SC 27,* p. 40, 1999.

[37] Rogaway, P.: Nonce-based symmetric encryption. In Roy, B., Meier, W. (eds.), *Fast Software Encryption*, vol. 3017 of *Lecture Notes in Computer Science*, pp. 348–358. Springer Berlin Heidelberg, 2004.

[38] Rogaway, P.: Authenticated-encryption with associated-data. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, CCS '02, pp. 98–107. ACM, New York, NY, USA, 2002.

[39] Namprempre, C., Rogaway, P., Shrimpton, T.: AE5 security notions: Definitions implicit in the CAESAR call. *Cryptology ePrint Archive, Report 2013/242*, 2013. `http://eprint.iacr.org/`.

[40] Selmer, E. S.: *Linear recurrence relations over finite fields*. University of Bergen, 1966.

[41] Rueppel, R. A.: *Analysis and Design of Stream Ciphers*. Communications and Control Engineering Series. Springer Berlin Heidelberg, 1986.

[42] Shannon, C. E.: A mathematical theory of communication. In *Bell system technical journal*, vol. 27, pp. 623–656. University of Illinois Press, 1948.

[43] Barker, E., Kelsey, J.: Recommendation for random number generation using deterministic random bit generators. *NIST Special Publication 800-90A,* 2012.

[44] Merkle, R.: One way hash functions and DES. In Brassard, G. (ed.), *Advances in Cryptology — CRYPTO' 89 Proceedings*, vol. 435 of *Lecture Notes in Computer Science*, pp. 428–446. Springer New York, 1990.

[45] Damgård, I.: A design principle for hash functions. In Brassard, G. (ed.), *Advances in Cryptology — CRYPTO' 89 Proceedings*, vol. 435 of *Lecture Notes in Computer Science*, pp. 416–427. Springer New York, 1990.

[46] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the indifferentiability of the sponge construction. In Smart, N. (ed.), *Advances in Cryptology – EUROCRYPT 2008*, vol. 4965 of *Lecture Notes in Computer Science*, pp. 181–197. Springer Berlin Heidelberg, 2008.

[47] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Cryptographic sponge functions. *Submission to NIST (Round 2),* 2011. `http://sponge.noekeon.org/CSF-0.1.pdf`.

[48] Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. In Menezes, A., Vanstone, S. (eds.), *Advances in Cryptology-CRYPT0'90*, vol. 537 of *Lecture Notes in Computer Science*, pp. 2–21. Springer Berlin Heidelberg, 1991.

[49] Alekseychuk, A., Schevtsov, A.: Upper estimates of imbalance of bilinear approximations for round functions of block ciphers. In *Cybernetics and Systems Analysis*, vol. 46, pp. 376–385. Springer US, 2010.

[50] Kovalchuk, L., Bezditnyi, V.: Upper bounds for the average probabilities of difference characteristics of block ciphers with alternation of Markov transformations and generalized Markov transformations. In *Cybernetics and Systems Analysis*, vol. 50, pp. 386–393. Springer US, 2014.

[51] Carlet, C.: *Vectorial Boolean functions for cryptography*. Boolean Models and Methods in Mathematics, Computer Science, and Engineering. Cambridge University Press, 2010.

[52] Matsui, M., Yamagishi, A.: A new method for known plaintext attack of FEAL cipher. In Rueppel, R. A. (ed.), *Advances in Cryptology — EUROCRYPT' 92*, vol. 658 of *Lecture Notes in Computer Science*, pp. 81–91. Springer Berlin Heidelberg, 1993.

[53] Nyberg, K.: Linear approximation of block ciphers. In De Santis, A. (ed.), *Advances in Cryptology — EUROCRYPT'94*, vol. 950 of *Lecture*

Notes in Computer Science, pp. 439–444. Springer Berlin Heidelberg, 1995.

[54] Matsui, M.: Linear cryptanalysis method for DES cipher. In Helleseth, T. (ed.), *Advances in Cryptology — EUROCRYPT '93*, vol. 765 of *Lecture Notes in Computer Science*, pp. 386–397. Springer Berlin Heidelberg, 1994.

[55] Bard, G. V.: *Algebraic cryptanalysis*. Springer, 2009.

[56] Canteaut, A.: Open problems related to algebraic attacks on stream ciphers. In Ytrehus, Ø. (ed.), *Coding and Cryptography*, vol. 3969 of *Lecture Notes in Computer Science*, pp. 120–134. Springer Berlin Heidelberg, 2006.

[57] Albrecht, M.: *Algorithmic algebraic techniques and their application to block cipher cryptanalysis*. Ph.D. thesis, Royal Holloway, University of London, the United Kingdom, 2010.

[58] Courtois, N., Pieprzyk, J.: Cryptanalysis of block ciphers with overdefined systems of equations. *Cryptology ePrint Archive, Report 2002/044*, 2002. `http://eprint.iacr.org/`.

[59] Weinmann, R.-P.: *Evaluating algebraic attacks on the AES*. Ph.D. thesis, Darmstadt University of Technology, Germany, 2003.

[60] Kleiman, E.: *High performance computing techniques for attacking reduced version of AES using XL and XSL methods*. Ph.D. thesis, Iowa State University, USA, 2010.

[61] Bulygin, S., Brickenstein, M.: Obtaining and solving systems of equations in key variables only for the small variants of AES. In *Mathematics in Computer Science*, vol. 3, pp. 185–200. Birkhäuser-Verlag, 2010.

[62] Courtois, N., Bard, G.: Algebraic cryptanalysis of the Data Encryption Standard. In Galbraith, S. (ed.), *Cryptography and Coding*, vol. 4887 of *Lecture Notes in Computer Science*, pp. 152–169. Springer Berlin Heidelberg, 2007.

[63] Kazymyrov, O., Oliynykov, R.: Construction of an overdefined system of equations describing the cipher "Labyrinth". In *Applied Radio Electronics*, vol. 8, pp. 247–251. Kharkiv National University of Radioelectronics, 2009. (In Russian).

[64] Kazymyrov, O., Oliynykov, R.: Algebraic properties of Kalyna's key schedule. In *Radioelectronic and computer systems*, vol. 5, pp. 61–66. National Aerospace University, Ukraine, 2010. (In Russian).

[65] Nyberg, K.: Differentially uniform mappings for cryptography. In Helleseth, T. (ed.), *Advances in Cryptology - EUROCRYPT'93*, vol. 765 of *Lecture Notes in Computer Science*, pp. 55–64. Springer Berlin Heidelberg, 1994.

[66] Biham, E.: New types of cryptanalytic attacks using related keys. In Helleseth, T. (ed.), *Advances in Cryptology — EUROCRYPT '93*, vol. 765 of *Lecture Notes in Computer Science*, pp. 398–409. Springer Berlin Heidelberg, 1994.

[67] Biryukov, A., Khovratovich, D.: Related-key cryptanalysis of the full AES-192 and AES-256. In Matsui, M. (ed.), *Advances in Cryptology – ASIACRYPT 2009*, vol. 5912 of *Lecture Notes in Computer Science*, pp. 1–18. Springer Berlin Heidelberg, 2009.

[68] Bogdanov, A., Khovratovich, D., Rechberger, C.: Biclique cryptanalysis of the full AES. In Lee, D., Wang, X. (eds.), *Advances in Cryptology – ASIACRYPT 2011*, vol. 7073 of *Lecture Notes in Computer Science*, pp. 344–371. Springer Berlin Heidelberg, 2011.

[69] Phan, R.-W.: Advanced slide attacks revisited: Realigning slide on DES. In Dawson, E., Vaudenay, S. (eds.), *Progress in Cryptology – Mycrypt 2005*, vol. 3715 of *Lecture Notes in Computer Science*, pp. 263–276. Springer Berlin Heidelberg, 2005.

[70] Biryukov, A., Wagner, D.: Slide attacks. In Knudsen, L. (ed.), *Fast Software Encryption*, vol. 1636 of *Lecture Notes in Computer Science*, pp. 245–259. Springer Berlin Heidelberg, 1999.

[71] Saarinen, M.-J.: A chosen key attack against the secret S-boxes of GOST. *Unpublished manuscript*, 1998.

[72] Kazymyrov, O.: Practical recovery of long-term keys of the GOST 28147 cipher based on slide attack. In *Science and social problems: Computerization and information technology*, pp. 272–273. Kharkiv National University of Radioelectronics, 2011. (In Russian).

[73] Isobe, T., Shibutani, K.: All subkeys recovery attack on block ciphers: Extending meet-in-the-middle approach. In Knudsen, L., Wu, H. (eds.), *Selected Areas in Cryptography*, vol. 7707 of *Lecture Notes in Computer Science*, pp. 202–221. Springer Berlin Heidelberg, 2013.

[74] Oliynykov, R.: *Methods for analysis and synthesis of perspective symmetric cryptographic transformations*. A thesis for a doctor of technical sciences degree in the specialty 05.13.05 – computer systems and components, Kharkiv National University of Radio Electronics, Ukraine, 2014. (In Russian).

[75] Zhou, Y., Feng, D.: Side-channel attacks: Ten years after its publication and the impacts on cryptographic module security testing. *Cryptology ePrint Archive, Report 2005/388*, 2005. `http://eprint.iacr.org/`.

[76] Nikova, S., Rechberger, C., Rijmen, V.: Threshold implementations against side-channel attacks and glitches. In Ning, P., Qing, S., Li, N. (eds.), *Information and Communications Security*, vol. 4307 of *Lecture Notes in Computer Science*, pp. 529–545. Springer Berlin Heidelberg, 2006.

[77] Kazymyrov, O., Kazymyrova, V., Oliynykov, R.: A method for generation of high-nonlinear S-boxes based on gradient descent. In *Mathematical Aspects of Cryptography*, vol. 5, pp. 71–78. Steklov Mathematical Institute, 2014.

[78] Nyberg, K.: Perfect nonlinear S-boxes. In Davies, D. (ed.), *Advances in Cryptology - EUROCRYPT'91*, vol. 547 of *Lecture Notes in Computer Science*, pp. 378–386. Springer Berlin Heidelberg, 1991.

[79] Chabaud, F., Vaudenay, S.: Links between differential and linear cryptanalysis. In *Advances in Cryptology—EUROCRYPT'94*, pp. 356–365. Springer, 1995.

[80] Budaghyan, L.: *Construction and analysis of cryptographic functions*. Habilitation Thesis, University of Paris 8, France, 2013.

[81] Kazymyrov, O., Oliynykov, R.: Choosing substitutions for symmetric cryptographic algorithms based on the analysis of teir algebraic properties. In *Mathematical modeling. Information Technology. Automated control systems.*, vol. 925, pp. 79–86. V. N. Karazin Kharkov National University, Ukraine, 2010. (In Russian).

[82] Budaghyan, L., Kazymyrov, O.: Verification of restricted EA-equivalence for vectorial Boolean functions. In Özbudak, F., Rodríguez-Henríquez, F. (eds.), *Arithmetic of Finite Fields*, vol. 7369 of *Lecture Notes in Computer Science*, pp. 108–118. Springer Berlin Heidelberg, 2012.

[83] Carlet, C., Charpin, P., Zinoviev, V.: Codes, bent functions and permutations suitable for DES-like cryptosystems. In *Designs, Codes and Cryptography*, vol. 15, pp. 125–156. Kluwer Academic Publishers, 1998.

[84] Krause, M.: BDD-based cryptanalysis of keystream generators. In Knudsen, L. (ed.), *Advances in Cryptology — EUROCRYPT 2002*, vol. 2332 of *Lecture Notes in Computer Science*, pp. 222–237. Springer Berlin Heidelberg, 2002.

[85] Stegemann, D.: Extended BDD-based cryptanalysis of keystream generators. In Adams, C., Miri, A., Wiener, M. (eds.), *Selected Areas in Cryptography*, vol. 4876 of *Lecture Notes in Computer Science*, pp. 17–35. Springer Berlin Heidelberg, 2007.

[86] Schilling, T., Raddum, H.: Solving compressed right hand side equation systems with linear absorption. In Helleseth, T., Jedwab, J. (eds.), *Sequences and Their Applications – SETA 2012*, vol. 7280 of *Lecture Notes in Computer Science*, pp. 291–302. Springer Berlin Heidelberg, 2012.

[87] Cid, C., Murphy, S., Robshaw, M.: Small scale variants of the AES. In Gilbert, H., Handschuh, H. (eds.), *Fast Software Encryption*, vol. 3557 of *Lecture Notes in Computer Science*, pp. 145–162. Springer Berlin Heidelberg, 2005.

[88] Shirai, T., Shibutani, K.: On the diffusion matrix employed in the Whirlpool hashing function. *NESSIE public reports,* 2003. `https://www.cosic.esat.kuleuven.be/nessie/nessie/reports/phase2/whirlpool-20030311.pdf`.

[89] Augot, D., Finiasz, M.: Direct construction of recursive MDS diffusion layers using shortened BCH codes. *In Pre-proceedings of Fast Software Encryption (FSE 2014),* 2014.

[90] Stein, W., et al.: Sage mathematics software (version 6.2). *The Sage Development Team,* 2014. `http://www.sagemath.org`.

[91] Biryukov, A., De Cannière, C., Braeken, A., Preneel, B.: A toolbox for cryptanalysis: Linear and affine equivalence algorithms. In Biham, E. (ed.), *Advances in Cryptology — EUROCRYPT 2003*, vol. 2656 of *Lecture Notes in Computer Science*, pp. 33–50. Springer Berlin Heidelberg, 2003.

[92] Burnett, L.: *Heuristic optimization of Boolean functions and substitution boxes for cryptography*. Ph.D. thesis, Queensland University of Technology, Australia, 2005.

[93] Lafitte, F., Heule, D. V., Hamme, J. V.: Cryptographic Boolean functions with R. In *The R Journal*, vol. 3, pp. 44–47. June, 2011.

[94] Mishra, P. R.: Calculating cryptographic degree of an S-box. *Cryptology ePrint Archive, Report 2014/145*, 2014. `http://eprint.iacr.org/`.

[95] Chmiel, K.: Fast computation of approximation tables. In Saeed, K., Pejaś, J. (eds.), *Information Processing and Security Systems*, pp. 125–134. Springer US, 2005.

[96] Yu, Y., Wang, M., Li, Y.: Constructing differential 4-uniform permutations from know ones. *Cryptology ePrint Archive, Report 2011/047*, 2011. `http://eprint.iacr.org/`.

[97] Tesař, P.: A new method for generating high non-linearity S-boxes. In *Radioengineering*, vol. 19, pp. 23–26. Brno University of Technology, 2010. `http://www.radioeng.cz/fulltexts/2010/10_01_023_026.pdf`.

[98] Matyukhin, D., Rudskoy, V., Shishkin, V.: A perspective hashing algorithm. *Materials of XII scientific conference RusCrypto'2010,* 2010. `http://www.ruscrypto.ru/resource/summary/rc2010/ruscrypto_2010_054.zip`. (In Russian).

[99] AlTawy, R., Youssef, A. M.: Integral distinguishers for reduced-round Stribog. *Cryptology ePrint Archive, Report 2013/648*, 2013. `http://eprint.iacr.org/`.

[100] Kazymyrov, O., et al.: Source code of the cross-platform implementation of Stribog. *GitHub repository,* 2013. `https://github.com/okazymyrov/stribog`.

[101] Kazymyrov, O., Leontiev, S., Popov, V., Smyshlyaev, S.: On creating effective software implementations of national cryptographic

standards. *Materials of XV scientific conference RusCrypto,* 2013. http://www.ruscrypto.ru/accotiation/archive/rc2013. (In Russian).

[102] Guo, J., Nikolic, I., Peyrin, T., Wang, L.: Cryptanalysis of Zorro. *Cryptology ePrint Archive, Report 2013/713*, 2013. http://eprint.iacr.org/.

[103] Hong, J., Kim, W.-H.: TMD-tradeoff and state entropy loss considerations of streamcipher MICKEY. In Maitra, S., Veni Madhavan, C., Venkatesan, R. (eds.), *Progress in Cryptology - INDOCRYPT 2005*, vol. 3797 of *Lecture Notes in Computer Science*, pp. 169–182. Springer Berlin Heidelberg, 2005.

[104] Jansen, C., Helleseth, T., Kholosha, A.: Cascade jump controlled sequence generator and Pomaranch stream cipher. In Robshaw, M., Billet, O. (eds.), *New Stream Cipher Designs*, vol. 4986 of *Lecture Notes in Computer Science*, pp. 224–243. Springer Berlin Heidelberg, 2008.

[105] Jansen, C.: The state space structure of the MICKEY stream cipher. *Proceedings of the 32rd WIC Symposium on Information Theory in the Benelux and The 1st Joint WIC/IEEE Symposium on Information Theory and Signal Processing in the Benelux,* 2011.

[106] Jansen, C.: Analysis of the nonlinear function of the Mickey S-register. *Proceedings of the 33rd WIC Symposium on Information Theory in the Benelux and The 2nd Joint WIC/IEEE Symposium on Information Theory and Signal Processing in the Benelux,* pp. 60–67, 2012.

[107] Olshansky, D.: Introduce GOST R 34.11-2012 hash function. *Electronic source,* 2014. http://rt.openssl.org/Ticket/Display.html?id=3311.